



Программный комплекс Систэм Платформ

---

SePlatform.HMI.IntegrityControl 2.1

---

Руководство пользователя

Редакция  
2. Предварительная

Соответствует версии ПО  
2.1.1

---



© ООО «СИСТЭМ СОФТ», 2022-2023. Все права защищены.

Авторские права на данный документ принадлежат ООО «СИСТЭМ СОФТ». Копирование, перепечатка и публикация любой части или всего документа не допускается без письменного разрешения правообладателя.

# Содержание

---

<b>1. О продукте</b>	<b>4</b>
<b>2. Установка и удаление</b>	<b>5</b>
<b>3. Дополнительная настройка окружения</b>	<b>7</b>
<b>4. Встраивание в проект</b>	<b>8</b>
4.1. Подключение (обновление) модуля SePlatform.HMI.IntegrityControl	8
4.2. Добавление экземпляра IntegrityControl в проект	9
4.3. API	11
4.3.1. Свойства	12
4.3.2. Команды	13
<b>5. Запуск в виде самостоятельного приложения</b>	<b>15</b>
OC Windows	15
OC Linux	15
<b>6. Возможности приложения</b>	<b>16</b>
6.1. Подключение к узлу	16
6.2. Проверка целостности	16
6.3. Создание эталона	17
6.4. Фильтрация файлов	18
6.5. Экспорт в файл	18
<b>История изменений</b>	<b>20</b>
2.0	20
2.0.1	20
Изменения документации	20
Редакция 1	20
Редакция 2	21
Редакция 3	21
1.2	21
1.2.1	22
Изменения документации	22
Редакция 1	22

# 1. О продукте

---

SePlatform.HMI.IntegrityControl - приложение для ручного запуска контроля целостности файлов и папок на локальных/удаленных узлах и просмотра результатов проверки. Приложение используется как дополнение к компонентам SePlatform.Security и SePlatform.HMI.Security, т.к. контроль целостности выполняется с помощью этих компонентов.

SePlatform.HMI.IntegrityControl можно запускать как отдельное приложение, либо встраивать его в прикладные проекты, разработанные в SePlatform.HMI. Целевыми проектами могут быть проекты, работающие на ОС Windows, ОС Linux, а также доступные через веб-интерфейс.

## 2. Установка и удаление

### ОС Windows

Для установки или удаления SePlatform.HMI.IntegrityControl используйте дистрибутив `seplatform.hmi.integritycontrol-<lng>-<version>.<arch>.msi`.



#### ПРИМЕЧАНИЕ

В названии файла `<lng>` - это язык компонента, `<version>` - номер версии компонента, а `<arch>` - целевая процессорная архитектура.

### ОС Linux

Для установки или удаления SePlatform.HMI.IntegrityControl используйте пакет `seplatform.hmi.integritycontrol-<lng>-<version>.<arch>.rpm` или `seplatform.hmi.integritycontrol-<lng>-<version>.<arch>.deb` в зависимости от выбранного пакетного менеджера ОС.



#### ПРИМЕЧАНИЕ

В названии пакета `<lng>` - это язык компонента, а `<version>` - номер версии компонента.



#### ОБРАТИТЕ ВНИМАНИЕ

Команды установки/удаления нужно запускать с правами суперпользователя **root**, находясь в папке с установочным пакетом.



#### ОБРАТИТЕ ВНИМАНИЕ

В командах удаления вместо полного имени пакета указывайте имя `seplatform.hmi.integritycontrol`.

### Пакетный менеджер YUM

#### Установка

```
yum install seplatform.hmi.integritycontrol-<lng>-<version>.<arch>.rpm
```

#### Удаление

```
yum remove seplatform.hmi.integritycontrol
```

### Пакетный менеджер RPM

#### Установка

```
rpm -i seplatform.hmi.integritycontrol-<lng>-<version>.<arch>.rpm
```

### Удаление

```
rpm -e seplatform.hmi.integritycontrol
```

## Пакетный менеджер APT

### Установка

```
apt-get install seplatform.hmi.integritycontrol-<lng>-<version>.<arch>.deb
```

### Удаление

```
apt-get remove seplatform.hmi.integritycontrol
```

## Пакетный менеджер DPKG

### Установка

```
dpkg -i seplatform.hmi.integritycontrol-<lng>-<version>.<arch>.deb
```

### Удаление

```
dpkg -r seplatform.hmi.integritycontrol
```

## 3. Дополнительная настройка окружения

---

Перед использованием SePlatform.HMI.IntegrityControl:

1. Если вы планируете подключаться к удаленным узлам, обязательно объедините свой компьютер и удаленные узлы в сеть SePlatform.Net. Настройте службу SePlatform.Net.Agent на каждом узле, включая узел с установленным SePlatform.HMI.IntegrityControl. Как организовать сеть SePlatform.Net описано в руководстве пользователя на SePlatform.Domain.
2. На тех компьютерах, где будет выполняться контроль целостности, в конфигурационных файлах SePlatform.Security:
  - Включите режим контроля целостности в файле `seplatform.security.agent.xml`.
  - Укажите контролируемые файлы и папки в файле `seplatform.security.ic.xml`.
  - Если нужно, включите периодическую автоматическую проверку контроля целостности в файле `seplatform.security.ic.xml`.

Подробнее о том, как настраивать SePlatform.Security, описано в руководстве пользователя на SePlatform.Security.

## 4. Встраивание в проект

Приложение SePlatform.HMI.IntegrityControl разработано средствами SePlatform.HMI и выполнено в виде проекта SePlatform.HMI. В собственные прикладные проекты SePlatform.HMI.IntegrityControl встраивается как отдельная библиотека - внешний модуль (подробнее о том, что такое внешний модуль, см. в руководстве пользователя на SePlatform.HMI).

Чтобы встроить SePlatform.HMI.IntegrityControl в проект:

1. Подключите SePlatform.HMI.IntegrityControl к проекту как внешний модуль ([стр. 8](#)).
2. Добавьте экземпляр типа **IntegrityControl** в проект ([стр. 9](#)).
3. Укажите для экземпляра типа **IntegrityControl** узел SePlatform.Net, на котором выполняется контроль целостности.



### ПРИМЕЧАНИЕ

Если узел локальный, его имя указывать не обязательно.

В интерфейсе встроенного экземпляра типа **IntegrityControl** вы увидите только дерево контролируемых файлов с указанного вами узла и результаты проверок по каждому файлу и папке. Если в SePlatform.Security включен периодический контроль целостности, вы также увидите как меняется окраска измененных файлов и папок.

Возможно, вы хотите изменить настройки типа **IntegrityControl** по умолчанию, добавить дополнительные кнопки, панель со статистикой и т.д. Используйте для этого API ([стр. 11](#)), которое предоставляет экземпляр типа **IntegrityControl**.

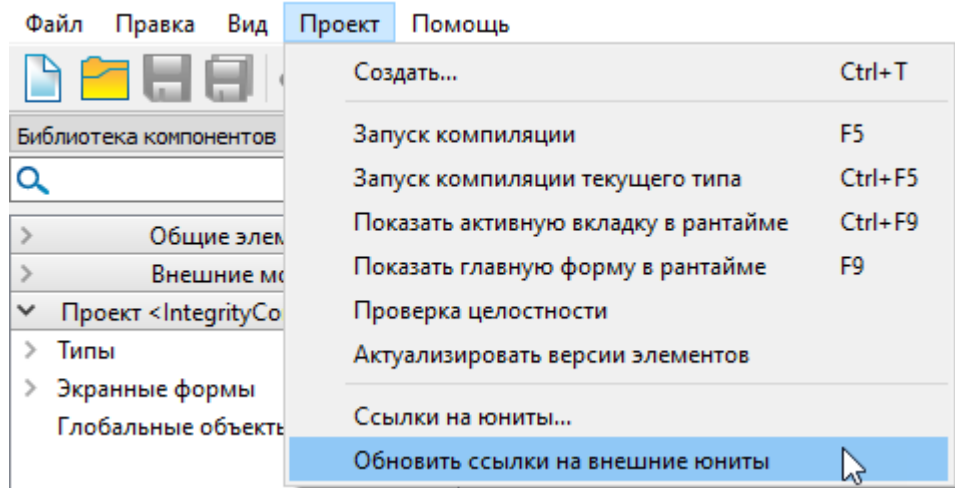
### 4.1. Подключение (обновление) модуля SePlatform.HMI.IntegrityControl

#### Подключить внешний модуль SePlatform.HMI.IntegrityControl к проекту

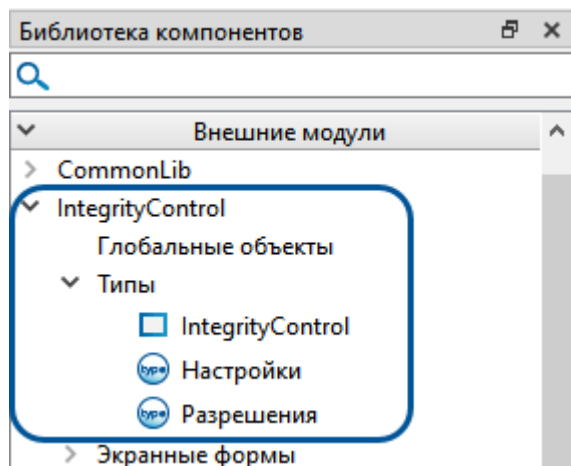
1. Создайте (если её там нет) в папке своего проекта папку `externals`, в которой нужно размещать файлы всех подключаемых внешних модулей.
2. Перейдите к папке, в которую устанавливаются все приложения SePlatform.HMI:
  - `C:\Program Files\SePlatform\SePlatform.HMI.Extensions` (ОС Windows);
  - `/opt/SePlatform/SePlatform.HMI.Extensions` (ОС Linux).
3. В папке уже должна быть папка `IntegrityControl`, появившаяся после установки SePlatform.HMI.IntegrityControl. Скопируйте эту папку `IntegrityControl` в созданную вами папку `externals`.
4. Откройте свой проект в дизайнере SePlatform.HMI.



5. Перейдите в меню **Проект** и выберите команду **Обновить ссылки на внешние юниты**.



Так вы обновите список внешних модулей своего проекта и новый модуль **IntegrityControl** появится в библиотеке компонентов.



## Обновить встроенный модуль SePlatform.HMI.IntegrityControl

Чтобы обновить уже встроенный в проект SePlatform.HMI.IntegrityControl:

1. Установите нужную версию SePlatform.HMI.IntegrityControl.
2. Обновите файлы внешнего модуля **IntegrityControl** в папке **externals** своего проекта. Для этого замените существующую папку **IntegrityControl** на папку **IntegrityControl** из папки установки SePlatform.HMI.IntegrityControl:

- C:\Program Files\SePlatform\SePlatform.HMI.Extensions (OC Windows);
- /opt/SePlatform/SePlatform.HMI.Extensions (OC Linux).

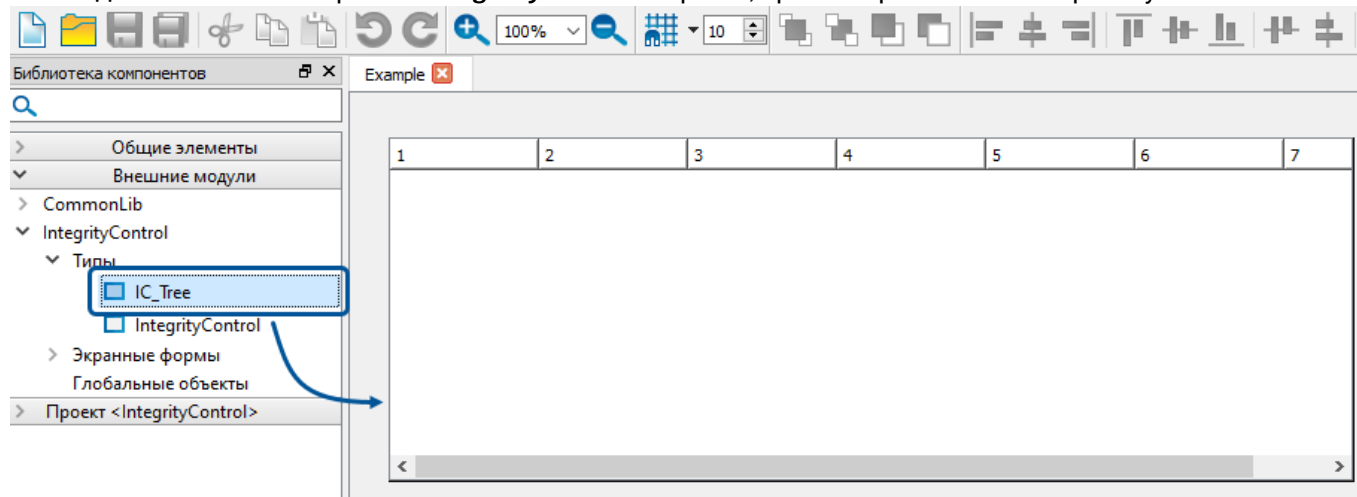
После обновления файлов модуля все экземпляры типа **IntegrityControl**, уже добавленные в проект, обновятся автоматически.

## 4.2. Добавление экземпляра IntegrityControl в проект

### Добавить экземпляр типа IntegrityControl в проект

Основной элемент в составе подключенного внешнего модуля **IntegrityControl** - тип **IntegrityControl**.

Чтобы добавить экземпляр типа **IntegrityControl** в проект, просто перетяните тип на рабочую область.



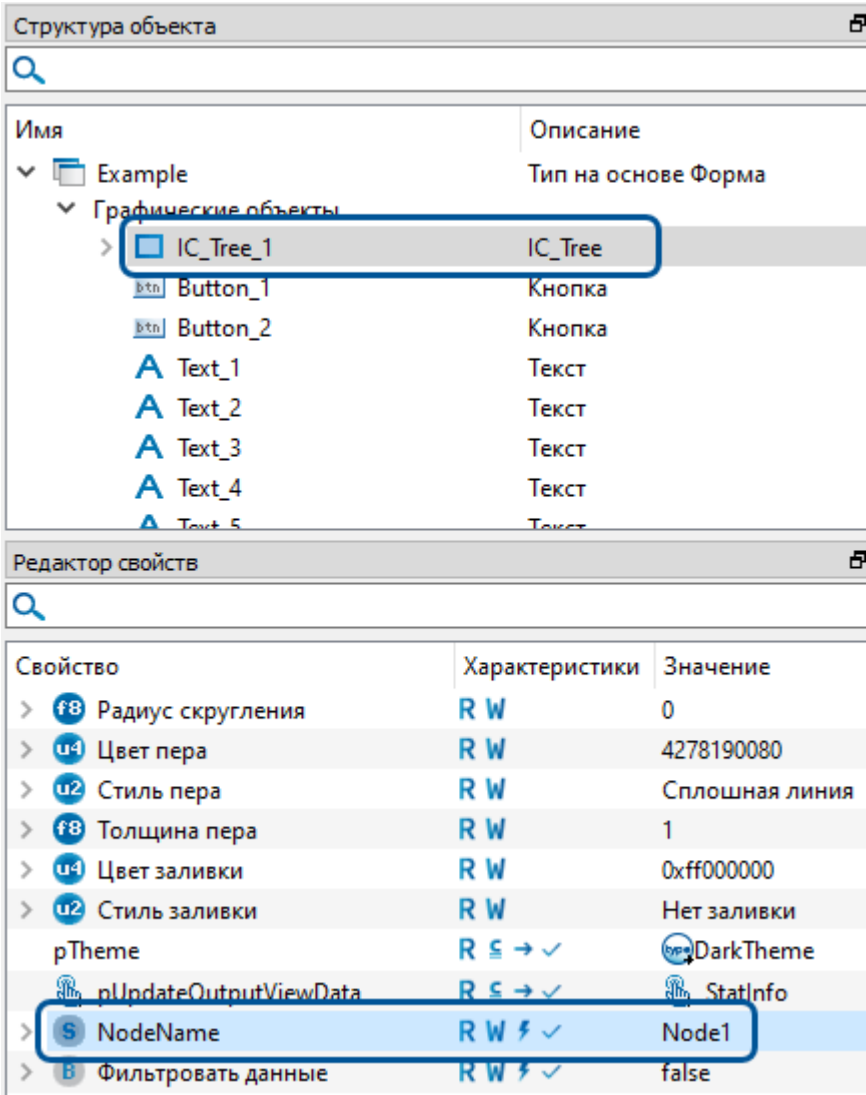
Вы можете добавить в проект сколько угодно экземпляров типа **IntegrityControl** и настроить каждый экземпляр под отдельные цели. Например, использовать разные экземпляры для просмотра контролируемых файлов и папок на разных узлах.

## Подключить экземпляр типа **IntegrityControl1** к узлу **SePlatform.Net**

Подключать **SePlatform.HMI.IntegrityControl** к узлу **SePlatform.Net** нужно только если вы хотите работать с удаленным узлом. Если же узел локальный, вы можете явно не указывать имя узла, т.к. **SePlatform.HMI.IntegrityControl** по умолчанию подключается к локальному узлу.

Чтобы подключить экземпляр типа **IntegrityControl** к узлу **SePlatform.Net**:

- 1. Перейдите к свойствам экземпляра (область **Редактор свойств**).
- 2. Найдите свойство **Имя узла сети** и укажите имя, которое указано для узла в файле конфигурации SePlatform.Net.Agent.



ОБРАТИТЕ ВНИМАНИЕ

Если вам не удалось получить данные с указанного узла, проверьте, что в SePlatform.Security включен контроль целостности и указанный узел входит в сеть SePlatform.Net ([стр. 7](#)).

### 4.3. API

Основной элемент добавляемого модуля SePlatform.HMI.IntegrityControl - это тип IntegrityControl1. Этот тип предоставляет API, которое можно использовать ([стр. 1](#)) для своих задач.

#### Свойства

Имя узла сети (nodeName) ( <a href="#">стр. 12</a> )	Имя узла SePlatform.Net.
Показывать только файлы с нарушением целостности (ShowOnlyChanged) ( <a href="#">стр. 12</a> )	Включает фильтрацию элементов дерева.

<b>Состояние (Status)</b> (стр. 12)	Текущий статус проверки целостности.
<b>Ошибка (Error)</b> (стр. 13)	Текст ошибки при выполнении контроля целостности.

## Команды

<b>Check</b> (стр. 13)	Выполняет проверку контролируемых файлов на узле, указанном в свойстве <b>Имя узла сети</b> (стр. 12).
<b>CreateEtalon</b> (стр. 14)	Создаёт эталон на узле, указанном в свойстве <b>Имя узла сети</b> (стр. 12).

### 4.3.1. Свойства

#### Имя узла сети (NodeName)

Имя узла SePlatform.Net, которое указано для узла в файле конфигурации SePlatform.Net. Можно не указывать значение, если узел локальный.

```
string NodeName
```

#### Показывать только файлы с нарушением целостности (ShowOnlyChanged)

Фильтрует список контролируемых файлов и папок еще до отображения их в SePlatform.HMI.IntegrityControl.

```
bool ShowOnlyChanged
```

#### Значение

« <b>true</b> »	В дереве контролируемых файлов отображаются только элементы с нарушением целостности, изменённой датой и т.д.
« <b>false</b> »	В дереве контролируемых файлов отображаются все элементы.

#### Состояние (Status)

Текущий статус проверки целостности.

```
string Status
```

## Ошибка (Error)

Текст ошибки при выполнении контроля целостности.

```
string Status
```

## Поля для вывода статистики

Тип `IntegrityControl1` предоставляет также поля, с помощью которых можно получить статистику создания эталона, выполнения проверки, а также узнать количество проверенных и поврежденных файлов. Такие поля недоступны в редакторе свойств, но к ним можно обратиться в любом обработчике через объекты `Information` и `Statistic`:

### ➤ Information

Группа текстовых полей с информацией о выполнении проверки и данных об эталонном значении.

EtalonDateCreated	string	Дата и время создания эталонного файла.
EtalonNameCreator	string	Имя пользователя, создавшего эталон.
LastCheckData	string	Дата и время выполнения последней проверки целостности файлов.

### ➤ Statistic

Группа полей для статистики в зависимости от статуса файла.

all	int4	Количество всех проверенных файлов.
violations	int4	Количество файлов с нарушением целостности (новые, удаленные и измененные файлы).
warnings	int4	Количество предупреждений.
errors	int4	Количество ошибок.



#### ПРИМЕР

Вывести в текстовые поля дату/время создания эталона и имя пользователя, создавшего эталон.

```
Text_1 = IntegrityControl_1.Information.EtalonDateCreated;
Text_7 = IntegrityControl_1.Information.EtalonNameCreator;
```

## 4.3.2. Команды

### Check

Выполняет проверку контролируемых файлов на узле, указанном в свойстве [Имя узла сети \(стр. 12\)](#).



## ПРИМЕР

Запустить контроль целостности на выбранном узле.

```
IntegrityControl_1.Check.Invoke();
```

## CreateEtalon

Создает эталон на узле, указанном в свойстве [Имя узла сети](#) (стр. 12).



## ПРИМЕР

Запустить контроль целостности на выбранном узле.

```
IntegrityControl_1.CreateEtalon.Invoke();
```

## 5. Запуск в виде самостоятельного приложения

---

### ОС Windows

Для запуска SePlatform.HMI.IntegrityControl как отдельного приложения воспользуйтесь командой Пуск → SePlatform → SePlatform.HMI.IntegrityControl.

### ОС Linux



#### ОБРАТИТЕ ВНИМАНИЕ

Запускать с правами администратора.

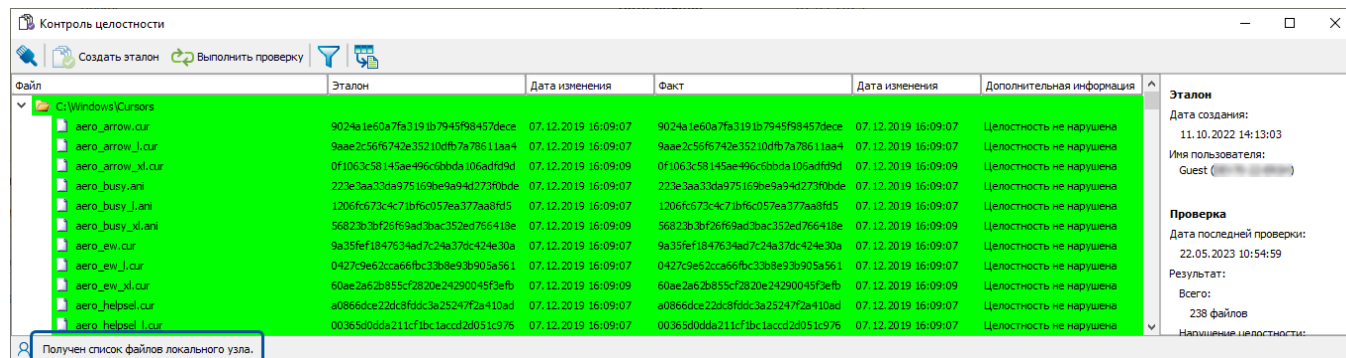
Для запуска SePlatform.HMI.IntegrityControl как отдельного приложения воспользуйтесь командой Пуск → Разработка → SePlatform.HMI.IntegrityControl (ОС Astra Linux). В других ОС семейства Linux команда открытия может быть другая.

## 6. Возможности приложения

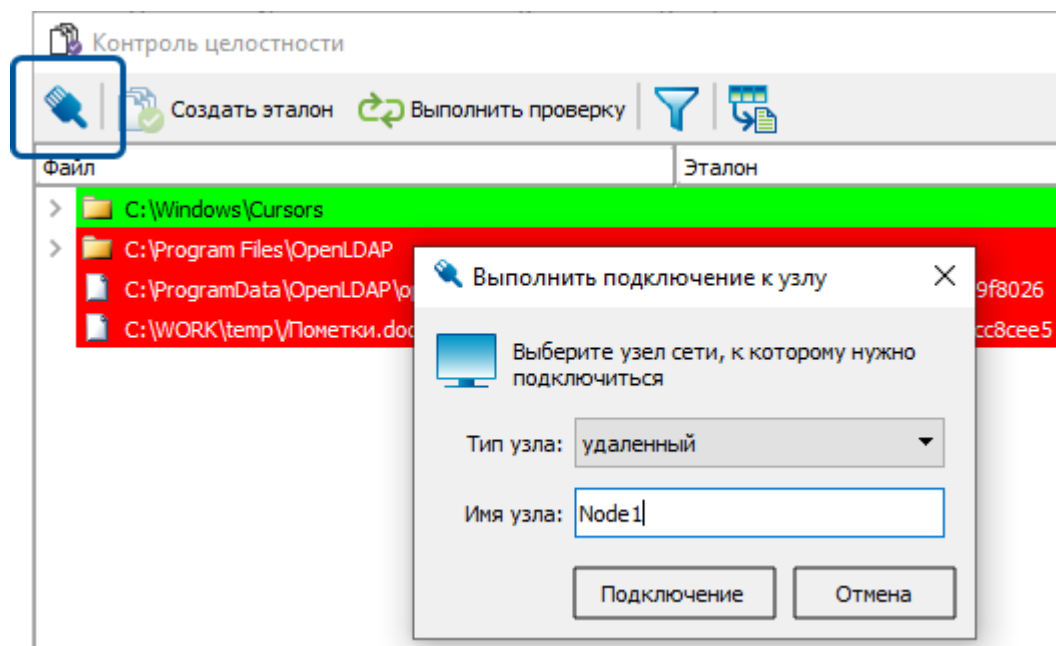
Если вы пользуетесь формой IntegrityControl1 или SePlatform.HMI.IntegrityControl, запущенным как отдельное приложение, вы видите уже готовый набор кнопок.

### 6.1. Подключение к узлу

По умолчанию приложение подключено к вашему локальному компьютеру, на котором и выполняется контроль целостности.



Чтобы подключиться к любому удаленному узлу (входящему вместе с вашим локальным узлом в сеть SePlatform.Net), нажмите кнопку **Выполнить подключение к узлу**, выберите тип узла и введите имя.



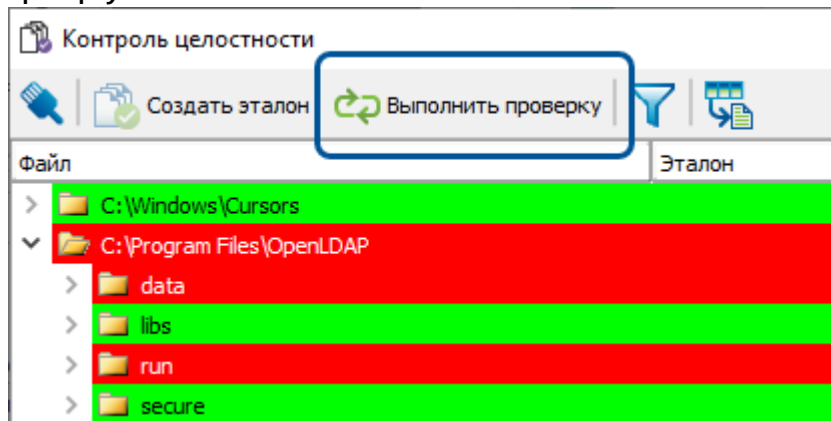
### 6.2. Проверка целостности

При проверке целостности проверяется соответствие контрольных сумм файлов и эталонных контрольных сумм. Список контролируемых файлов ([стр. 7](#)) формируется заранее в конфигурационном файле службы SePlatform.Security.

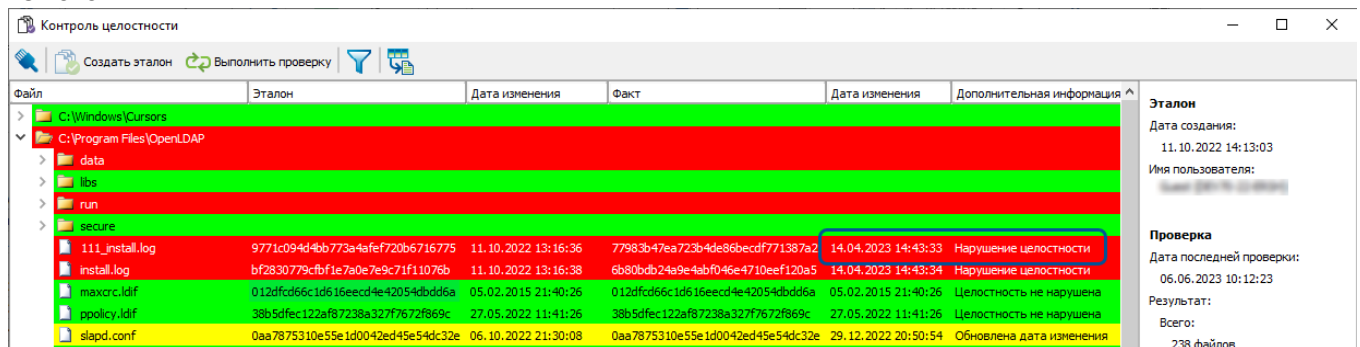


Проверка может выполняться автоматически (стр. 7) с заданным периодом, если это указано при настройке службы SePlatform.Security.

Чтобы вручную запустить проверку целостности на подключенном узле, нажмите кнопку **Выполнить проверку**.

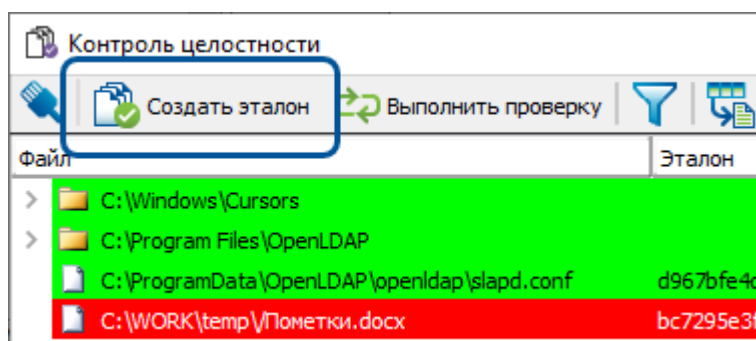


В результате проверки по каждому файлу вы можете отследить был ли изменен файл и когда он был изменен.



## 6.3. Создание эталона

Под эталоном понимается срез текущего состояния (контрольных сумм) по каждому файлу, с которым будет сравниваться состояние файлов в при следующей проверке. Чтобы создать эталон, нажмите кнопку **Создать эталон**.



Для каждого файла вы можете увидеть эталонную контрольную сумму и дату изменения эталона.

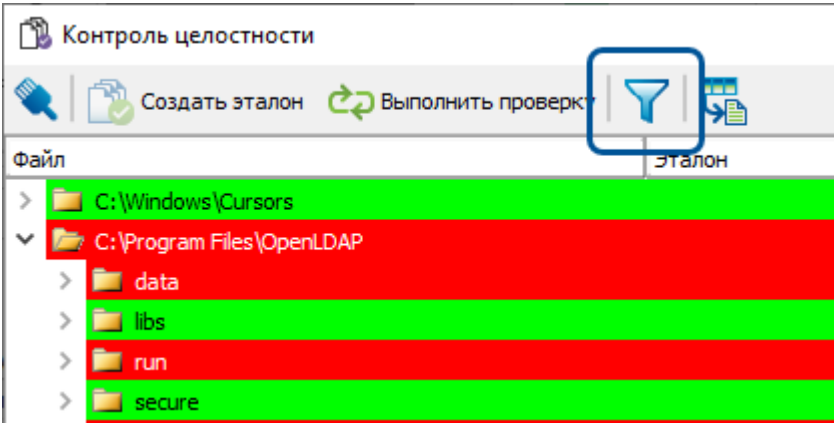
Контроль целостности

Создать эталонВыполнить проверку

Файл	Эталон	Дата изменения	Факт	Дата изменения	Дополнительная информация	Эталон
C:\Windows\Cursors						Дата создания: 11.07.2022 15:22:41
C:\Program Files\OpenLDAP						Имя пользователя: Guest(DEV70-22-ERSH)
C:\ProgramData\OpenLDAP\openldap\slapd.conf	d067bfe4d892a554cab72700e02f345f	08.07.2022 15:24:44	d967bfe4d892a554cab73799e92f345f	08.07.2022 15:24:44	Целостность не нарушена	
C:\WORK\temp\Пометки.docx	bc7295e3f0570f12f08f66eedb940124	01.10.2021 11:05:32	1efc8a837385a81dba6d9458e5c8cb58	11.07.2022 15:22:56	Нарушение целостности	Проверка

6.4. Фильтрация файлов

Чтобы отобразить только файлы с изменениями, нажмите кнопку **Показать файлы с нарушением целостности**. Чтобы снова отобразить все файлы, нажмите кнопку еще раз.



6.5. Экспорт в файл

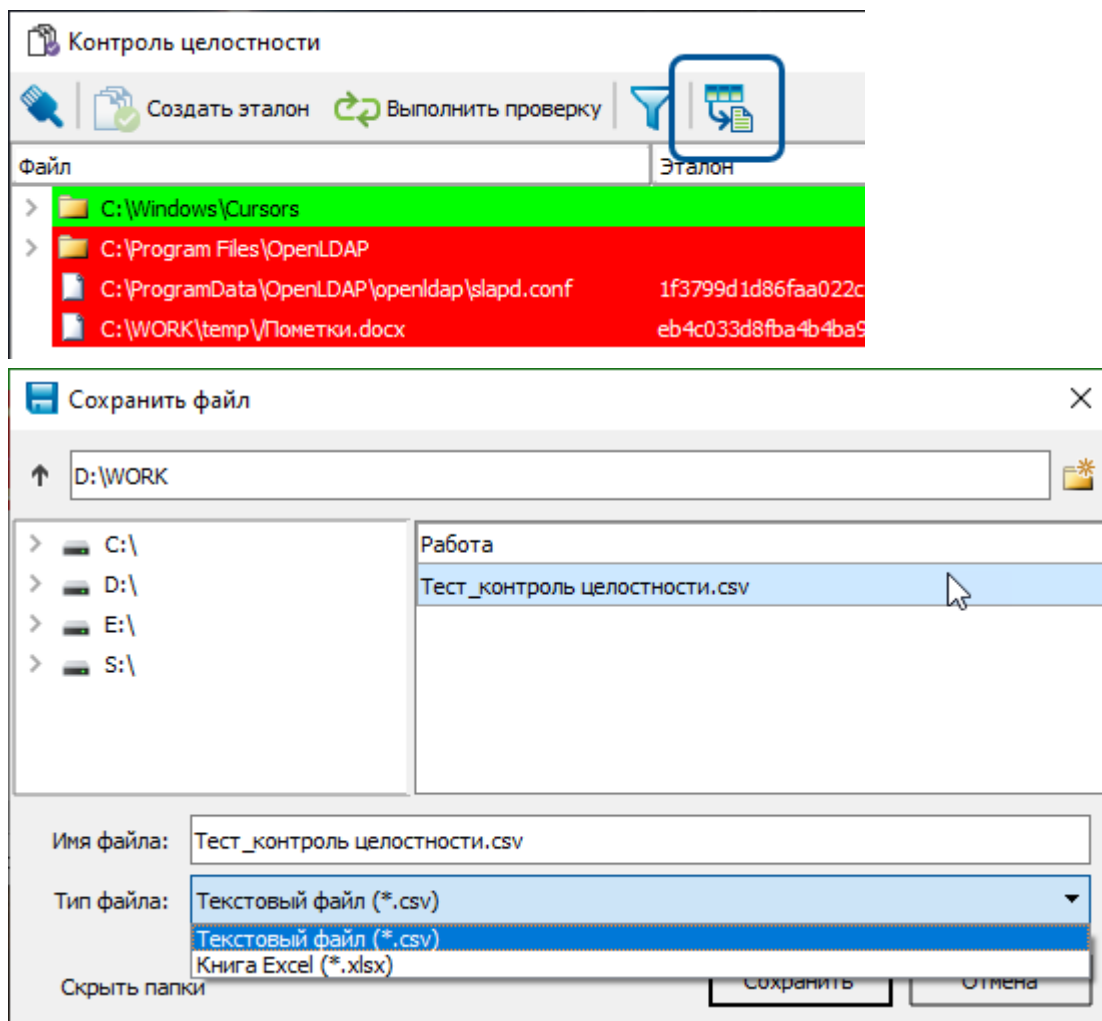
Вы можете сохранить информацию о контролируемых файлах в формате таблицы (\*.csv или \*.xlsx).

Номер	Каталог	Файл и расширение	Эталонная контрольная сумма	Текущая контрольная сумма	Эталонная дата и время	Текущая дата и время	Состояние
1	C:\Windows\Cursors						
1.1		C:\Windows\Cursors\aero_arrow...	9024a1e60a7fa3191b7945f98457dece	9024a1e60a7fa3191b7945f98457dece	07.12.2019 16:09:07	07.12.2019 16:09:07	Целостность не нарушена
1.2		C:\Windows\Cursors\aero_arrow_...	9aae2c56f6742e35210dfb7a78611aa4	9aae2c56f6742e35210dfb7a78611aa4	07.12.2019 16:09:07	07.12.2019 16:09:07	Целостность не нарушена
1.3		C:\Windows\Cursors\aero_arrow_...	0f1063c58145ae496c6bbda106adfd9d	0f1063c58145ae496c6bbda106adfd9d	07.12.2019 16:09:09	07.12.2019 16:09:09	Целостность не нарушена
1.4		C:\Windows\Cursors\aero_busy...	223e3aa33da975169be9a94d273f0bde	223e3aa33da975169be9a94d273f0bde	07.12.2019 16:09:07	07.12.2019 16:09:07	Целостность не нарушена
1.5		C:\Windows\Cursors\aero_busy_...	1206fc673c4c71bf6c057ea377aa8fd5	1206fc673c4c71bf6c057ea377aa8fd5	07.12.2019 16:09:07	07.12.2019 16:09:07	Целостность не нарушена

По каждому файлу в таблицу сохраняется:

- имя файла и расширение;
- эталонная контрольная сумма;
- текущая контрольная сумма;
- эталонная дата и время;
- текущая дата и время;
- текущее состояние файла (информация о существовании файла, нарушении целостности и обновлении даты изменения файла).

Чтобы сохранить информацию в файл, нажмите кнопку **Экспорт в файл**, выберите место сохранения и формат файла.



# История изменений

## 2.0

### Новая возможность

Для ограничения прав пользователей вместо свойств **Разрешить выполнять проверку** (**AllowCheck**) и **Разрешить создание эталонных значений** (**AllowCreateEtalon**) типа **IntegrityControl** теперь используйте новый тип Разрешения (**IntegrityControl\_Permissions**). Новый тип содержит свойства **Проверка файлов** (**CheckFiles**) и **Создание эталонных значений** (**CreateEtalon**).



#### ОБРАТИТЕ ВНИМАНИЕ

При переходе на новую версию SePlatform.HMI.IntegrityControl для использования нового типа Разрешения вам потребуется доработать проект.

В экземпляре типа **IntegrityControl** в параметре инициализации **Разрешения** сошлитесь на нужный экземпляр типа **Разрешения**. Если вы используете **IntegrityControl\_Form**, можете создать экземпляр разрешений прямо в обработчике открытия формы.

Если вы желаете использовать систему безопасности для ограничения прав пользователя:

1. Создайте в подсистеме безопасности приложение для SePlatform.HMI.IntegrityControl с правами:
  - CheckFiles (Проверка файлов);
  - CreateEtalon (Создание эталонных значений).
2. Для экземпляра типа Разрешения:
  - Включите использование подсистемы безопасности (в свойстве **Использовать сервер безопасности для определения прав** (**UseSecurity**)).
  - Укажите имя приложения, которое вы используете для SePlatform.HMI.IntegrityControl в подсистеме безопасности (в свойстве **Имя приложения с правами безопасности** (**SecurityApplication**)).

### Исправленная ошибка

Исправлена ошибка, из-за которой сообщения о состоянии файла не соответствовали сообщениям аудита.

## 2.0.1

Внутренние изменения. Функциональность компонента не менялась.

## Изменения документации

### Редакция 1

- Добавлено описание ([стр. 18](#)) возможности выгрузки информации о контролируемых файлах в таблицу.
- В описание настройки окружения ([стр. 7](#)) добавлены названия конфигурационных файлов SePlatform.Security.
- Описана возможность ([стр. 16](#)) выбора узла для подключения.

- Описано как запускать приложение в ОС Linux ([стр. 15](#)).
- Ошибочно указанная служба SePlatform.Net заменена на SePlatform.Security ([стр. 16](#)).

## Редакция 2

- Обновлено скриншоты в разделе с описанием встраивания SePlatform.HMI.IntegrityControl в проект ([стр. 8](#)).
- В нескольких разделах имя типа изменено с IC\_Tree на IntegrityControl.

## Редакция 3

Внутренние изменения. Содержимое документа не изменилось.

# 1.2

### Новые возможности

- Переработано API компонента, используемого для встраивания в проекты пользователя. Ранее этот компонент назывался IC\_Tree, теперь называется IntegrityControl. API компонента стало проще и понятней для использования.

Состав нового API:

- Свойства:
  - **NodeName** для задания имени узла сети.
  - **AllowCheck**, разрешающее выполнять проверку.
  - **AllowCreateEtalon**, разрешающее создавать эталонные значения.
  - **ShowOnlyChanged** для включения фильтрации элементов дерева и отображения только файлов с нарушением целостности.
  - **Status**, содержащее описание текущей операции.
  - **Error**, содержащее описание ошибки, которая произошла при выполнении последней операции.
- Команды:
  - **Check** для проверки контролируемых файлов на узле, указанном в свойстве **NodeName**.
  - **CreateEtalon** для создания эталона на узле, указанном в свойстве **NodeName**.
  - Добавлены объекты **Information** и **Statistic**, с помощью свойств которых можно получить дополнительную информацию о целостности системы.
- Добавлена форма **IntegrityControl\_Form** для открытия с определенными параметрами. Такая форма может использоваться, например, для контроля определенного узла.
  - Свойства формы:
    - **AllowCheck**, разрешающее выполнять проверку.
    - **AllowCreateEtalon**, разрешающее создавать эталонные значения.
  - Параметры инициализации:
    - **init\_NodeName** для задания имени узла сети.
    - **init\_RestoreWindow** для разрешения восстановления положения окна.
- В строке статуса теперь выводится текст ошибок подключения к узлу.
- Реализовано сохранение ширины столбцов в дереве файлов.

### Исправленные ошибки

Исправлена проблема, из-за которой не работали свойства **AllowCheck** (**Разрешить выполнять проверку**) и **AllowCreateEtalon** (**Разрешить создание эталонных значений**).

## 1.2.1

Новая возможность

Добавлена возможность экспорта данных из дерева в таблицу формата \*.csv и \*.xlsx.

Исправленная ошибка

Папка, не содержащая контролируемые файлы, ошибочно окрашивалась в желтый цвет.

## Изменения документации

### Редакция 1

Внутренние изменения. Содержимое документа не изменилось.